2011-01-08

Organization & Analysis of Stock Option Market Data

Jun Zhang
Worcester Polytechnic Institute

Organization & Analysis of Stock Option Market Data

A Professional Master's Project

Submitted to the Faculty of the WORCESTER

POLYTECHNIC INSTITUTE In partial fulfillment

of the requirements for the Professional Degree of

Master of Science

in

Financial Mathematics by

Jun Zhang

December 2010

Approved:

Professor Domokos Vermes, Advisor

Professor Bogdan Vernescu, Head of Department

Abstract

Option market data are quoted in terms of option prices and are fragmented into over 100 individual contract files per day for each symbol. Traders and quantitative analysts compare values of options in terms of implied volatilities. The current project refactors fragmented option price data into implied volatility files organized by stock symbols and expiration dates. Each resulting file comprises the temporal evolution of daily volatility smile curves for every day prior to expiration. Possible analysis enabled by the refactored data is demonstrated.

Executive Summary

Option market data contain valuable information on market participants' views regarding future price evolution of a particular security. Most of this information is complementary to the underlying security's current price and price history. In the current project we focus on stock options data.

The difficulty of accessing this quantitative information originates in the complicated structure of option data quotes. At any given time more than 100 option contracts are quoted on a typical heavily traded stock symbol. These are put and call contracts corresponding to at least three different expiration dates and approximately 10 different strike prices. Apart from the most recently transacted option price, the quotes contain bid and ask prices, daily volumes and open interest data. Not all contracts are actively traded, consequently "most recent" prices may be stale and not related to the current stock price.

Option prices expressed in dollars are difficult to compare due to the changing price of the underlying security vs. the fixed grid of strike prices. For this reason traders are not evaluating options in terms of their quoted dollar prices but in terms of their implied volatilities.

Implied volatilities expressed as function of the moneyness ratio (strike price/ current stock price) of their contact exhibit the well-known "smile curve" pattern. Far out-of-the-money contracts sell at a premium as compared to their in-the-money siblings. This is a consequence of the fact that stock returns and prices have heavier tailed probability distributions than the normal distribution, on which the Black-Scholes option pricing theory is based.

The primary objective of the present project is to reorganize daily option market price data in such a format that is more amenable to quantitative analysis and which is based on implied volatilities.

We organize data according to stock symbols and option expiration dates. This means that each single file contains all prior dates and strike prices corresponding to the same expiration date and stock symbol. Hence each file contains a sequence of daily smile curves for each day prior to the expiration date for the given stock symbol. We also preserve trading volume and bid-ask spread data in similarly structured but separate parallel files.

We use our own fully documented algorithm to convert option prices into implied volatilities. The algorithm assures that the implied volatilities of at-the-money put and call options coincide and hence the resulting smile curves have no discontinuities at moneyness = 1.

The data reorganization and conversion is implemented in two stages, first by a compiled C program for speed and then an R script for the probabilistic-financial details. We explicitly construct all smile curve files for all stock symbols in the current S&P 100 index. Our programs are capable to produce similar files for arbitrary user-defined symbol and expiration date sets.

In the final section we demonstrate a variety of possible analysis of the information contained in the option market data that can be easily performed using our refactored implied volatility database.

Acknowledgements

It is my greatest pleasure to have this opportunity to give special thanks to all the people that have helped me during my graduate study at WPI. I would like to specially thank my advisor, Professor Vermes Domokos, for his guidance, support during my graduate study and eventually this master project. I would like to thank my friends, family, and wonderful wife, Hongmei Wang, for their emotional support over the past 3 years.

Contents

1. Background

Black-Scholes model is widely used to model the prices of equity options in the financial markets. There are 6 assumptions underlying the basic Black-Scholes model.

1. Option can only be exercised upon expiration (European options).

2. No commissions are charged in the transaction.

3. Interest rates remain constant and are known.

4. Stock pays no dividends. (This assumption can be relaxed.)

5. Stock prices move according to a geometric Brownian motion, i.e. stock returns follow a (generalized) Brownian motion with a possible drift term.

6. The volatility of the returns process is constant. This implies that the returns are normally distributed.

The Black-Scholes formula expresses the option price as a function of the stock price, the strike price, the time-to-expiration, interest rate and volatility. Assuming all other parameters being kept constant, the Black-Scholes formula establishes a one-to-one correspondence between volatility and the option price and hence can be inverted. Implied volatility (IV) is the volatility value, that would yield the given option price under the Black-Scholes model and assumptions.

Under Black-Scholes assumptions the implied volatility should be the same for all strike prices. But if we calculate the implied volatility based on the observed market prices of the options, then the resulting implied volatility will depend on the strike prices.

This disparity is known as the volatility skew. If we plot the implied volatilities (IV) against the strike prices (K) we get a U-shaped curve resembling a smile. Hence, this particular volatility skew pattern is better known as the volatility smile.

Several factors may contribute to the volatility smile pattern:

1. Options whose strike price is far away from the stock price are thinly traded less liquid. Hence a trader, who must buy or sell may not be negotiate a fair market price and have to pay a liquidity premium. Higher option price means higher implied volatility. In-the-money options are in general unattractive to traders as they tie up large amount of trading capital. Consequently liquidity premium often affects the entire in-the-money side of the option price curve. For this reason, in the present study we ignore in-the-money options and base all our curves and analyses on at-the-money and out-of-the-money option prices.
2. An option is an insurance against a gain/loss in the stock price. Insurance against a large loss/gain may cost relatively more. This means that option prices corresponding to a large absolute difference between stock price and strike price are relatively more expensive. This implies a higher volatility value for the far out options than the volatility implied by the at-the money options.
3. The stock return distribution cannot be modeled with the normal distribution model. The heavy tail distribution of the stock return is another factor that contributes the volatility smile curve. A heavy-tailed return distribution means that large deviations from the current stock price are more likely than what is predicted by the normal distribution. Insurance against these more likely extreme losses cost more, which translates to higher option prices and higher implied volatility at the outer ends of the strike price spectrum.

Put-call parity defines a relationship between the price of a call option and a put option—both with the identical strike price and expiration date. Consider a stock portfolio that contains a put option and a share. The portfolio value at expiration T will be K with S_T<=K or S_T with S_T >= K. Then consider a stock portfolio that contains a call option and zero coupon bonds K with face value K discounted at annual continuously compounded interest rate r. The portfolio value at expiry T will be K with S_T <=K or S_T with S_T >= K. Now that whatever the final share price S is at time T, each portfolio is worth the same as the other. This implies that these two portfolios must have the same value at any time t before T.

Thus the following relationship exists between the values of the various instruments at a general time *t*:

$$C(t)+K.\exp(-r.t) = P(t)+S(t)$$

Where C(t) is the value of the call at time t,

P(t) is the value of the put,

S(t) is the value of the share,

K is the strike price.

Put-call parity must hold for at-the-money options. In the present project we will use this relationship to determine the interest rate r implied by the market option prices.

2. Financial Data Description

The stock option trading data comes from an earlier master's project "Restructuring Option Chain Data Sets Using Matlab" by Alison Wooden completed at WPI in May 2010. It contains 60 different option trading data for approximately 4500 stock symbols with trading dates between 01/2005 and 12/2009. All the stock option trading files are saved with Comma Separated Values (.CSV) file format. A CSV file is simple text format for database table. Each field value of a record is separated from the next by a character that is mostly a comma. It has two different formats: by ticker or by date. The trading data by ticker format includes all the trading data from the same stock symbol. While the trading data by date format includes single trading day's data with all the different stock symbols. For every trading data, it contains the following information: stock symbol, stock trading price, option root, option extension, option type(put or call option), option expiration date, option trading date, strike price, option last trading price, option bid price, option ask price, option volume, option open interest, implied volatility, Greek Letters(including delta, gamma, theta). It is not known what algorithms and assumption (e.g. risk-free interest rate) were used in the conclusion of implied volatility and of the Greeks. For this reason, these data will not be used. The following table is a sample set of the option trading data file:

stock	price		optroot	optext	opttype	expiry	dataday	strike	optlast	bid	ask	volume	interest	iv	delta
CSCO	19.3	*	CYQ	AA	call	01/21/05	1/3/2005	5	14.4	14.3	14.4	5	5345	2.7924	0.994244
CSCO	19.3	*	CYQ	MA	put	01/21/05	1/3/2005	5	0.05	0	0.05	0	13249	2.774	-0.00555
CSCO	19.3	*	CYQ	AU	call	01/21/05	1/3/2005	7.5	11.8	11.8	11.9	2	1980	1.9745	0.992152
CSCO	19.3	*	CYQ	MU	put	01/21/05	1/3/2005	7.5	0.05	0	0.05	0	14096	1.9775	-0.00791
CSCO	19.3	*	CYQ	AB	call	01/21/05	1/3/2005	10	9.4	9.3	9.4	37	5380	1.4008	0.989622
CSCO	19.3	*	CYQ	MB	put	01/21/05	1/3/2005	10	0.05	0	0.05	0	10926	1.4175	-0.01105
CSCO	19.3	*	CYQ	AV	call	01/21/05	1/3/2005	12.5	6.9	6.8	6.9	10	8398	0.9552	0.98607
CSCO	19.3	*	CYQ	MV	put	01/21/05	1/3/2005	12.5	0.05	0	0.05	0	20744	0.9796	-0.01579
CSCO	19.3	*	CYQ	AC	call	01/21/05	1/3/2005	15	4.4	4.3	4.4	0	26649	0.5875	0.979649
CSCO	19.3	*	CYQ	MC	put	01/21/05	1/3/2005	15	0.05	0	0.05	0	36147	0.6126	-0.02455
CSCO	19.3	*	CYQ	AW	call	01/21/05	1/3/2005	17.5	1.95	1.9	1.95	268	45693	0.3899	0.887899
CSCO	19.3	*	CYQ	MW	put	01/21/05	1/3/2005	17.5	0.1	0.05	0.1	1153	68761	0.3672	-0.09931
CSCO	19.3	*	CYQ	AD	call	01/21/05	1/3/2005	20	0.25	0.2	0.25	5625	207083	0.2853	0.303711
CSCO	19.3	*	CYQ	MD	put	01/21/05	1/3/2005	20	0.85	0.8	0.9	1181	75283	0.2571	-0.71791
CSCO	19.3	*	CYQ	AX	call	01/21/05	1/3/2005	22.5	0.05	0	0.05	65	98610	0.3868	0.039453

Figure 1. Sample Option Trading Table

From the sample data, the following patterns can be observed:

1. For the trade-by-symbol option trading data, the option trading data is in the order of the expiration date and trading date. The call data is always followed with the put data with the same expiration and trading date. For every trading date, the trading data is ordered with the strike price.
2. For the trade-by-date option trading data, the option trading data is in the order of the stock symbol with the same option trading date. The call data is always followed with the put data with the same expiration and trading date. For every stock symbol, the trading data is ordered with the strike price.
3. The trading volume is the highest as the strike price gets closer to the stock price (At the money option). As the strike price gets farther away from the stock price, the trading volume decreases dramatically (Out of the money option).
4. The bid, ask and stock price always reflect the closing stock price. The final option trading price might not fall between the bid and ask price. In that case, the option trading price might not reflect the closing price of the option trading.
5. If the trading volume is very low or even reaches 0, the trading price does not reflect the real-time market price. It reflects an earlier trade that is not reflected with the current bid ask price.

6. The interest rate is not given. That makes it necessary to figure out how to get the interest rate to calculate the implied volatility.

3. Methodology

To analyze the huge amount of data set, the following methods are being used:

1. Define the stock symbols of the interest. In the project code, it defines stock symbol table with S&P 100 index symbols with an array.

```
stockSymbol<-
c("AA", "AAPL", "ABT", "AEP", "ALL", "AMGN", "AMZN", "AVP", "AXP", "BA",
  "BAC", "BAX",  "BHI", "BK", "BMY", "BRK.B", "CAT", "C", "CL", "CMCSA",
  "COF", "COP", "COST", "CPB", "CSCO", "CVS", "CVX", "DD", "DELL", "DIS",
  "DOW", "DVN", "EMC", "ETR", "EXC", "F",  "FCX", "FDX", "GD", "GE", "GILD",
  "GOOG", "GS", "HAL", "HD", "HNZ", "HON",  "HPQ", "IBM", "INTC", "JNJ",
  "JPM", "KFT", "KO", "LMT", "LOW", "MA", "MCD", "MDT", "MET",  "MMM",
  "MO", "MON", "MRK", "MS", "MSFT", "NKE", "NOV", "NSC",  "NWSA", "NYX",
  "ORCL", "OXY", "PEP", "PFE", "PG", "PM", "QCOM",  "RF", "RTN",  "S", "SLB",
  "SLE", "SO", "T", "TGT", "TWX", "TXN", "UNH", "UPS", "USB", "UTX", "VZ",
  "WAG", "WFC", "WMB",  "WMT", "WY", "XOM", "XRX");
```

   As the stock symbols come from the current S&P100 index and there was some recent addition or removal of the stock to the S&P 100 index, some stock symbol like brk.b does not have the corresponding data file. In addition, this array can be easily expanded or extended. If S&P 500 index data needs to be analyzed, simply replace this array with S&P 500 index array.

2. Define the stock option trading period. In the project code, it defines 60 different stock option expiration dates that spans from 01/2005 to 12/2009.

```
tradingDates<-
c("01-21-05", "02-18-05", "04-15-05", "03-18-05", "07-15-05", "05-20-05",
  "06-17-05", "08-19-05", "09-16-05", "10-21-05", "11-18-05", "12-16-05",
```

```
"01-20-06", "02-17-06", "03-17-06", "04-21-06",  "05-19-06", "06-16-06",
"07-21-06", "08-18-06", "10-20-06", "09-15-06", "11-17-06", "12-15-06",
"01-19-07", "02-16-07", "03-16-07", "04-20-07", "05-18-07", "06-15-07",
"07-20-07", "08-17-07", "09-21-07", "10-19-07", "11-16-07", "12-21-07",
"01-18-08", "02-15-08", "03-20-08", "04-18-08", "05-16-08", "06-20-08",
"07-18-08", "08-15-08", "09-19-08", "10-17-08", "11-21-08", "12-19-08",
"01-16-09", "02-20-09", "03-20-09", "04-17-09", "05-15-09", "06-19-09",
"07-17-09", "08-21-09", "09-18-09", "10-16-09", "11-20-09", "12-18-09");
```

This trading period array can be easily extended or adjusted as well.

3. For every stock symbol's comma separated value file, retrieve all the option trading data with the same expiration date and sort it out with the first part as the call trading data and the second part with the put trading data. Save the data into a new csv file for further processing. For example, if the stock symbol is AAPL and the expiration date is 01/25/2005, the option trading data information with the expiration date 01/25/2005 will be saved in the file AAPL_01-25-2005.csv. As 60 different option expiration dates were selected, there will be 60 different files per stock symbol. This utility function to perform this task is written with C language code. The following is a sample table file generated with the C utility function.

SLE	7.95	*	SLE	JA	call	10/16/09	02/23/09	5	0	2.9	3.2	0	0	0.5816	0.8908	5.0437	-0.1503	1.1936
SLE	7.95	*	SLE	JU	call	10/16/09	02/23/09	7.5	1.36	1.15	1.3	2	0	0.4284	0.6354	13.7502	-0.221	2.3969
SLE	7.95	*	SLE	JB	call	10/16/09	02/23/09	10	0	0.3	0.45	0	0	0.4231	0.3081	13.0371	-0.2034	2.2443
SLE	8.24	*	SLE	JA	call	10/16/09	02/24/09	5	0	3.1	3.4	0	0	0.5255	0.9195	4.3096	-0.1135	0.9857
SLE	8.24	*	SLE	JU	call	10/16/09	02/24/09	7.5	1.3	1.35	1.5	10	2	0.4335	0.6734	12.6085	-0.2232	2.3794
SLE	8.24	*	SLE	JB	call	10/16/09	02/24/09	10	0	0.4	0.55	0	0	0.4261	0.3475	13.1407	-0.2235	2.4373
SLE	7.85	*	SLE	JA	call	10/16/09	02/25/09	5	0	2.8	3	0	0	0.4836	0.9138	5.183	-0.1052	0.986
SLE	7.85	*	SLE	JU	call	10/16/09	02/25/09	7.5	1.1	1.1	1.2	20	12	0.4147	0.621	14.6285	-0.2148	2.3861
SLE	7.85	*	SLE	JB	call	10/16/09	02/25/09	10	0.25	0.25	0.4	7	0	0.4157	0.2884	13.0955	-0.1923	2.1415
SLE	7.76	*	SLE	JA	call	10/16/09	02/26/09	5	0	2.8	3.1	0	0	0.639	0.8688	5.3859	-0.1839	1.3172
SLE	7.76	*	SLE	JU	call	10/16/09	02/26/09	7.5	1.1	1.1	1.25	0	32	0.4599	0.6105	13.4811	-0.2376	2.3729
SLE	7.76	*	SLE	JB	call	10/16/09	02/26/09	10	0.35	0.25	0.4	44	7	0.4288	0.2856	12.811	-0.1955	2.1025
SLE	7.71	*	SLE	JA	call	10/16/09	02/27/09	5	0	2.75	3	0	0	0.5964	0.8756	5.6079	-0.165	1.2583
SLE	7.71	*	SLE	JU	call	10/16/09	02/27/09	7.5	1.1	1.05	1.2	0	32	0.4525	0.603	13.8917	-0.234	2.3649
SLE	7.71	*	SLE	JB	call	10/16/09	02/27/09	10	0.35	0.25	0.4	0	51	0.4365	0.2841	12.6612	-0.1977	2.0792
SLE	7.95	*	SLE	VA	put	10/16/09	02/23/09	5	0	0.2	0.3	0	0	0.6285	-0.12	4.9901	-0.1698	1.2762
SLE	7.95	*	SLE	VU	put	10/16/09	02/23/09	7.5	0	1	1.1	0	0	0.5378	-0.362	10.921	-0.2707	2.3899
SLE	7.95	*	SLE	VB	put	10/16/09	02/23/09	10	0	2.6	2.8	0	0	0.557	-0.613	10.7766	-0.2843	2.4428
SLE	8.24	*	SLE	VA	put	10/16/09	02/24/09	5	0	0.2	0.3	0	0	0.6563	-0.112	4.3963	-0.1753	1.256
SLE	8.24	*	SLE	VU	put	10/16/09	02/24/09	7.5	0	0.9	1	0	0	0.539	-0.331	10.1941	-0.2729	2.3918
SLE	8.24	*	SLE	VB	put	10/16/09	02/24/09	10	2.47	2.4	2.6	6	0	0.5478	-0.586	10.7774	-0.2958	2.5701
SLE	7.85	*	SLE	VA	put	10/16/09	02/25/09	5	0	0.2	0.3	0	0	0.6216	-0.123	5.2229	-0.1695	1.2771
SLE	7.85	*	SLE	VU	put	10/16/09	02/25/09	7.5	0	1	1.15	0	0	0.5457	-0.372	11.0521	-0.275	2.3724
SLE	7.85	*	SLE	VB	put	10/16/09	02/25/09	10	2.47	2.65	2.85	0	6	0.5545	-0.626	10.8946	-0.2774	2.3762
SLE	7.76	*	SLE	VA	put	10/16/09	02/26/09	5	0	0.2	0.35	0	0	0.6523	-0.134	5.3491	-0.1868	1.3355
SLE	7.76	*	SLE	VU	put	10/16/09	02/26/09	7.5	0	1	1.15	0	0	0.5324	-0.384	11.591	-0.2682	2.3622
SLE	7.76	*	SLE	VB	put	10/16/09	02/26/09	10	2.47	2.65	2.85	0	6	0.5311	-0.649	11.2808	-0.2571	2.2932
SLE	7.71	*	SLE	VA	put	10/16/09	02/27/09	5	0	0.2	0.35	0	0	0.6487	-0.136	5.4735	-0.1866	1.3358
SLE	7.71	*	SLE	VU	put	10/16/09	02/27/09	7.5	0	1	1.15	0	0	0.5254	-0.39	11.908	-0.2648	2.3536
SLE	7.71	*	SLE	VB	put	10/16/09	02/27/09	10	2.47	2.7	2.85	0	6	0.5177	-0.663	11.4955	-0.2454	2.2387

Figure 2. Sample table file generated with the C utility function

4. In the R code, read the sorted trading data file processed in the previous step and read the call/put trading data into separate call/put data vector, use this vector to get the information of interest including option price, option volume, option bid ask spread, trading date, trading interval before expiration.

5. With the call/put strike price K and stock price S, get the strike price that makes the moneyness K/S closest to 1, which we call ATM (At The Money). At this point, get the interest rate R with the call-put parity with the formula:

$$C(t)+K.\exp(-r.T) = P(t)+S(t)$$

6. With the newly calculated interest rate information and the rest of the option information, calculate the Implied Volatility (IV) corresponding to all strike prices with the Black-Scholes Model.

7. Save the trading date, days till expiration, interest rate, the implied volatility matrix with row name with strike price and column name with the stock price. The file is named as stock_name_date_call_iv.csv or stock_name_date_put_iv.csv. The following is the sample data for that file. Inside the file, if the market data does not yield a meaningful value for the interest rate, the interest rate will be set as 0 and all the implied volatility will be set as -1.

TradingDa	04/18/05	04/19/05	04/20/05	04/21/05	04/22/05	04/25/05	04/26/05	04/27/05	04/28/05
DaystoExp	43	42	41	40	39	38	37	36	35
Interest	0.023023	0	0.063355	0.03022	0.063271	0.031811	0.063978	0.042351	0.034828
StockPrice	35.62	37.09	35.51	37.18	35.5	36.98	36.19	35.95	35.54
22.5	0.083374	-1	0.083374	0.083374	0.083374	0.083374	0.960792	1.123627	1.309713
25	0.041626	-1	0.041626	0.041626	0.599071	0.083374	0.083374	0.041626	0.742942
27.5	0.498219	-1	0.953712	0.041626	0.559415	0.041626	0.041626	0.041626	0.624241
30	0.434044	-1	0.704087	0.041626	0.478151	0.403013	0.708592	0.531211	0.462479
32.5	0.449387	-1	0.606755	0.020874	0.441833	0.409323	0.397178	0.445186	0.411371
35	0.435146	-1	0.428258	0.422028	0.422979	0.420214	0.431316	0.429606	0.428952
37.5	0.432103	-1	0.399628	0.403703	0.36439	0.43261	0.413648	0.420095	0.413646
40	0.406924	-1	0.38998	0.38341	0.41261	0.416466	0.413999	0.419872	0.43068
42.5	0.41426	-1	0.42004	0.377249	0.417219	0.409815	0.409063	0.417478	0.429907
45	0.412719	-1	0.419053	0.363722	0.382541	0.417988	0.433597	0.434014	0.435721

Figure 3. Sample stock option implied volatility table

8. Save the bid-ask spread information into a file called stock_name_date_call_spread.csv or stock_name_date_put_spread.csv. The following table is a sample file for that.

TradingDa	04/18/05	04/19/05	04/20/05	04/21/05	04/22/05	04/25/05	04/26/05	04/27/05	04/28/05
DaystoExp	43	42	41	40	39	38	37	36	35
Interest	0.023023	0	0.063355	0.03022	0.063271	0.031811	0.063978	0.042351	0.034828
StockPrice	35.62	37.09	35.51	37.18	35.5	36.98	36.19	35.95	35.54
22.5	0.2	0.2	0.2	0.3	0.2	0.3	0.3	0.3	0.1
25	0.2	0.2	0.2	0.2	0.2	0.2	0.3	0.3	0.2
27.5	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.1
30	0.2	0.2	0.2	0.2	0.1	0.2	0.2	0.2	0.1
32.5	0.1	0.2	0.2	0.1	0.2	0.2	0.2	0.1	0.2
35	0.1	0.1	0.1	0.1	0.05	0.2	0.2	0.15	0.1
37.5	0.1	0.1	0.05	0.05	0.05	0.1	0.15	0.1	0.1
40	0.1	0.1	0.05	0.05	0.1	0.05	0.1	0.05	0.1
42.5	0.05	0.05	0.05	0.05	0.05	0.05	0.1	0.1	0.05

Figure 4. Sample stock option bid-ask spread table

9. Save the option trading volume information into a file called stock_name_date_call_vol.csv or stock_name_date_put_vol.csv. The following table is a sample file for that.

TradingDat	04/18/05	04/19/05	04/20/05	04/21/05	04/22/05	04/25/05
DaystoExpi	43	42	41	40	39	38
Interest	0.02302	0	0.06336	0.03022	0.06327	0.03181058
StockPrice	35.62	37.09	35.51	37.18	35.5	36.98
22.5	0	5	0	0	0	0
25	0	0	20	0	15	37
27.5	50	50	30	0	15	2
30	3	162	359	25	388	425
32.5	197	197	36	19	20	487
35	303	320	338	1526	1557	1301
37.5	222	222	789	419	631	177
40	423	425	262	78	28	103
42.5	37	37	20	0	10	35

Figure 5. Sample stock option trading volume table

4. Implementation

The implementation of the project includes two parts: The first part is a utility file that is written in C language, which parses individual stock option trading data and saves each the stock trading with the option expiration date specified into separate files. The binary file datacollect.exe is provided to convert individual stock option trading data. The following is the syntax of the command line:

datacollect -inputdir <directory name> -outputDir <directory name>

-sf <file name> -df <fileName>

-inputdir: Provide the directory information for the stock option trading data.

-outputdir: Provide the directory information for the converted data to place into.

-sf: Provide all the stock symbols to have the data converted.

-df: Provide all the option expiration dates.

The following is an example of the usage for the utility binary:

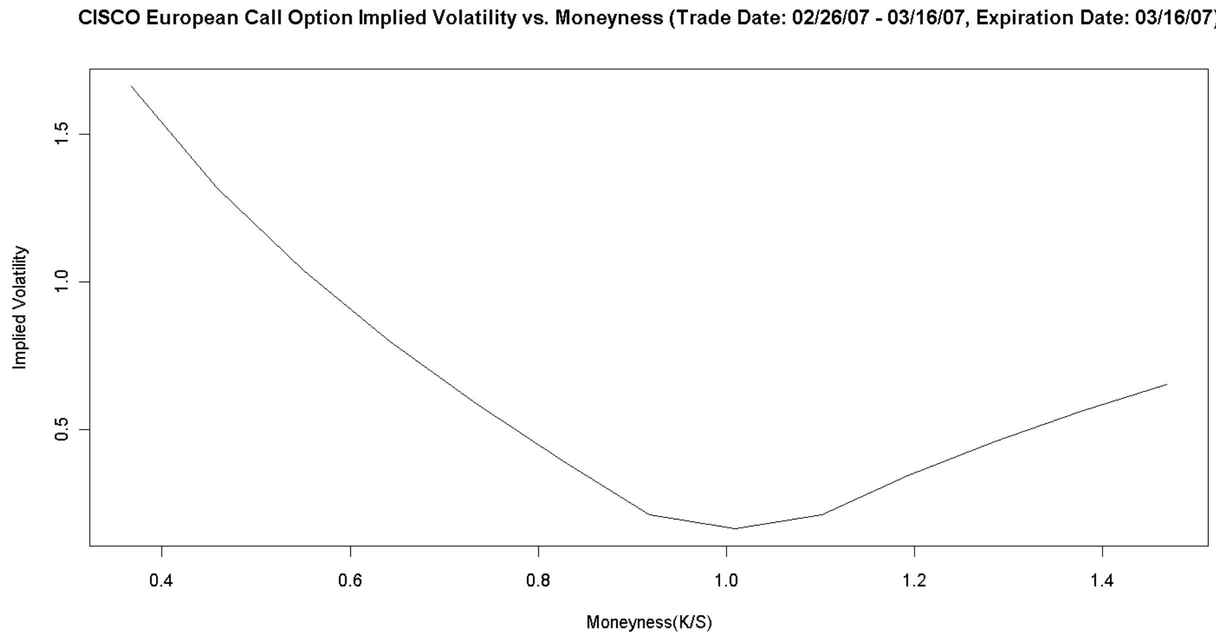
datacollect -inputdir "C:\finance data\optionsData\by Ticker" -outputDir c:\datacollect\output -df c:\datacollect\tradingdate.txt –sf c:\datacollect\stock.txt

The second stage of the data conversion and analysis is implemented by an R script. It uses the converted data files produced by the first stage and calculates the implied volatilities. The interest rate needed for the implied volatility calculation is determined from the put-call parity requirement. Implied volatilities are organized and written into . csv data files where columns corresponds to trading dates and rows to different moneyness (strike price) levels. Bid-ask spreads and trading volumes are written into identically organized separate files.

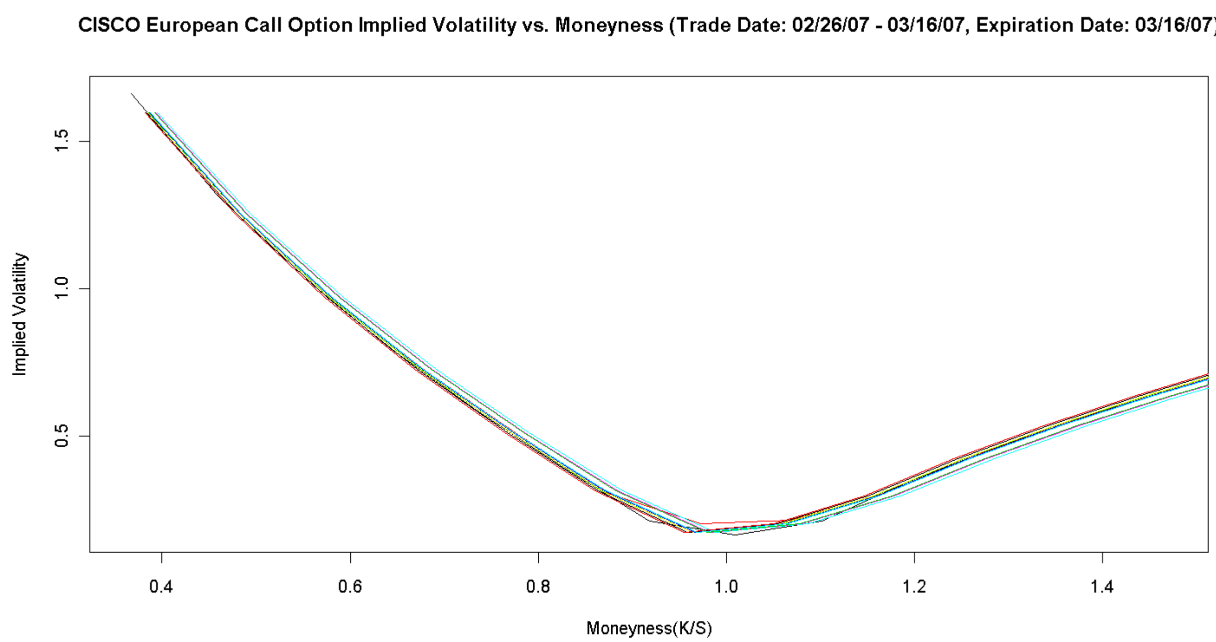
5. Analysis

The organization of the data achieved in the previous chapters of the present project makes various analysis and comparisons of option trading data possible. In the present chapter we give examples of such analysis possibilities. The full analysis of 60 contracts regarding approximately 4500 stock symbols is beyond the scope of the present project.

From the real market data, we can confirm the occurrence of the smirk curves. The following is the graph of the CISCO stock option Implied Volatility vs. Moneyness. It can be seen that the implied volatility is the lowest as the strike price is close to the stock price. The implied volatility increases as the option becomes increasingly in-the-money or out-of-the-mo

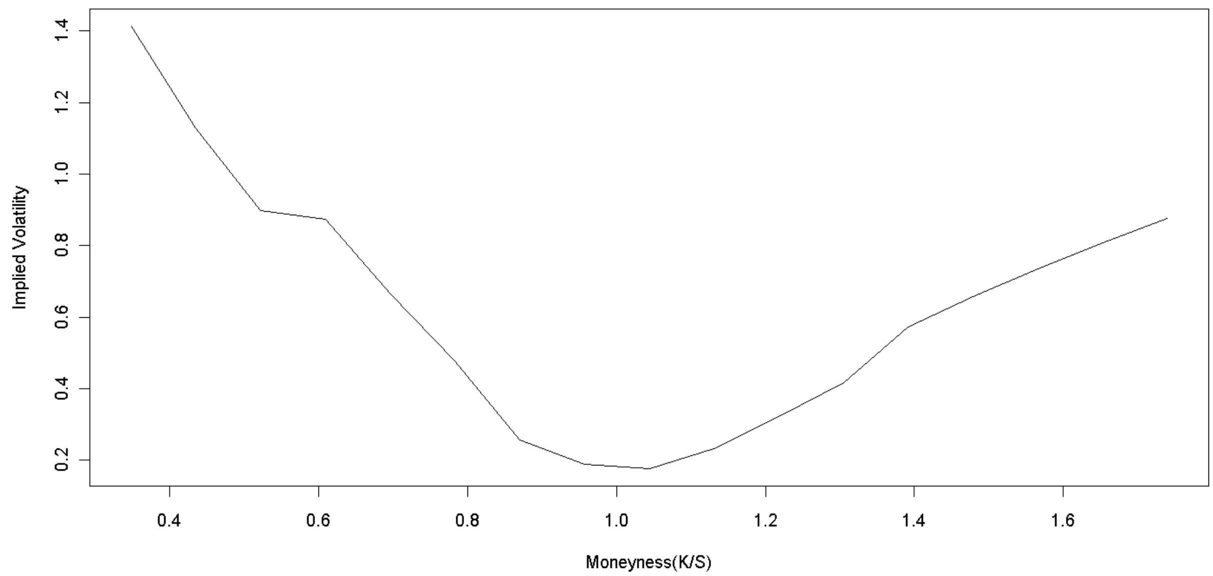


As the stock option gets closer to the expiration date, it can be seen that the shape of the curve remains consistent.



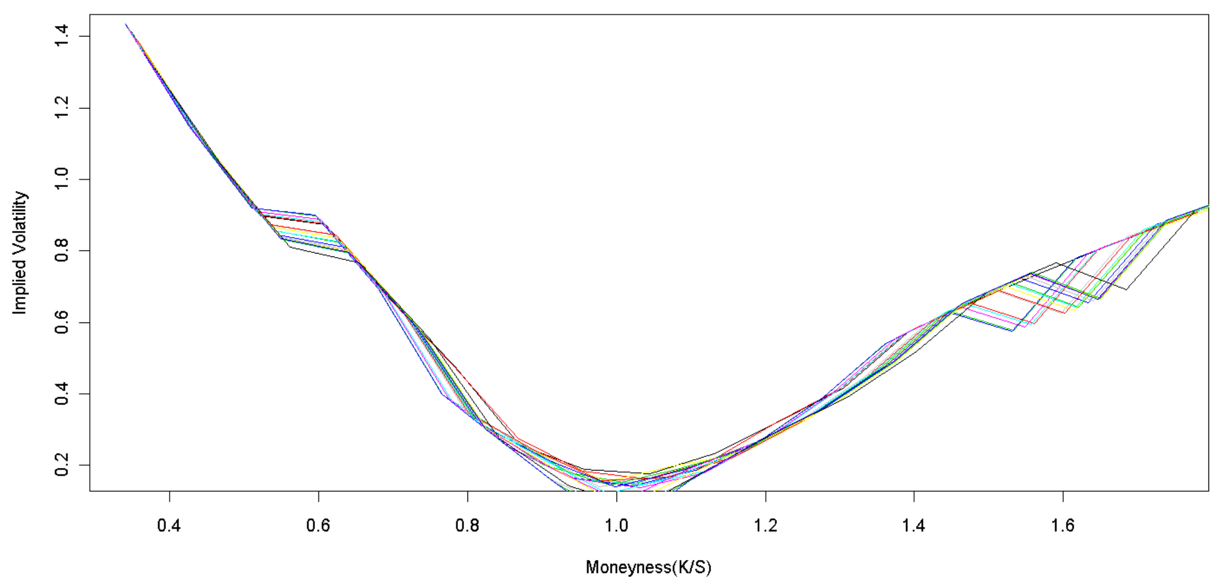
Here is another graph of the Microsoft stock option Implied Volatility vs. Moneyness.

MSFT European Call Option Implied Volatility vs. Moneyness (Trade Date: 02/26/07 - 03/16/07, Expiration Date: 03/16/07)

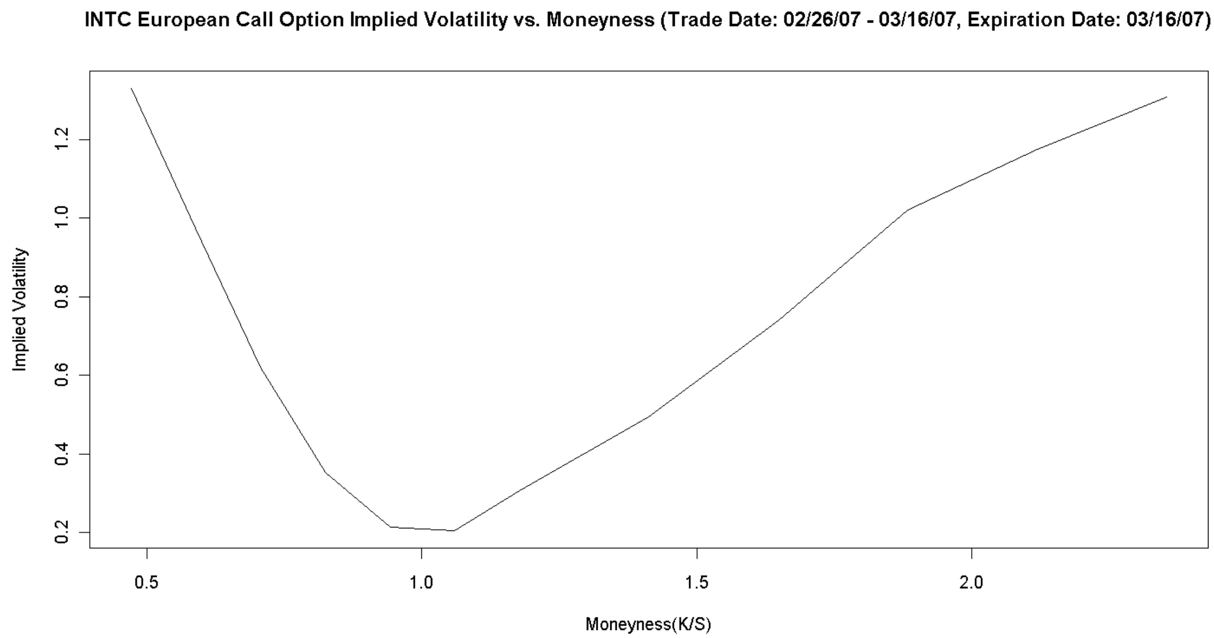


With the option trading dates change toward the option expiration date, it can be seen that the curve remains consistent.

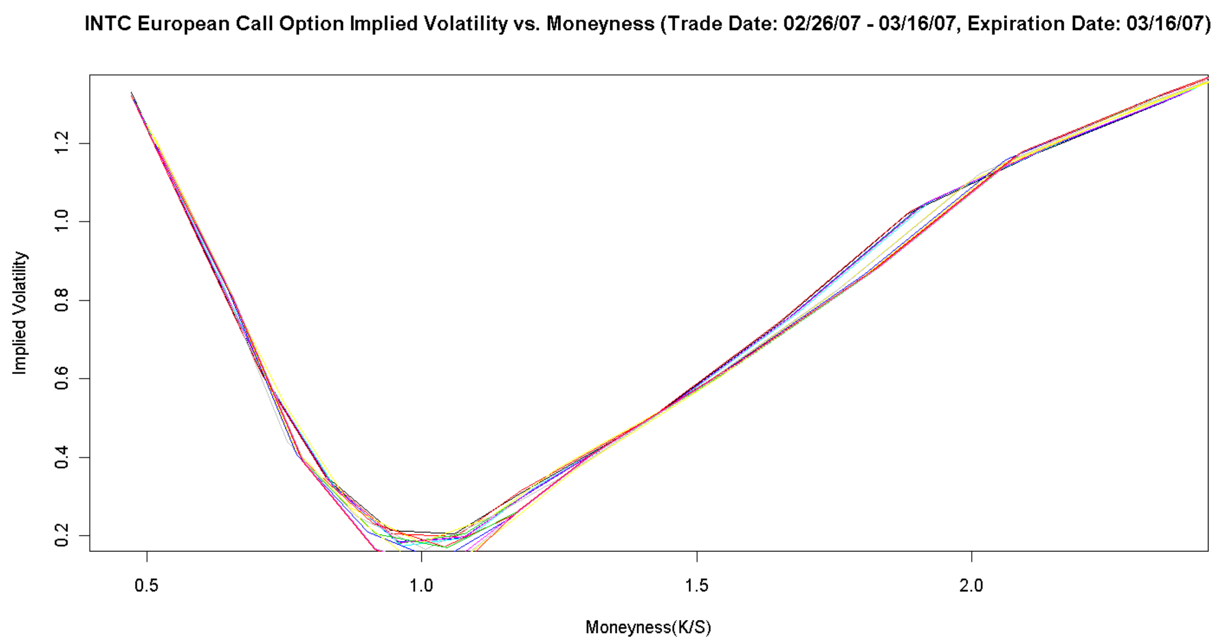
MSFT European Call Option Implied Volatility vs. Moneyness (Trade Date: 02/26/07 - 03/16/07, Expiration Date: 03/16/07)



The following curve displays the implied volatility extracted from out-of-the money options. It provides European Put option's (out-of-money) Implied Volatility when moneyness is below 1 and European Call option's (out-of-money) Implied Volatility when moneyness is above 1.



The next graph is the sequence of the dates that gets closer to the expiration date.



From the trading data, it can be seen that if moneyness gets close to one, there is usually sufficient trading volume and narrow bid-ask spread. So these implied volatility values can be considered as good quality. For moneyness below 0.8 and greater than 1.2, the trading volume is usually small, often zero and bid-ask spreads are larger. The implied volatility obtained in those regions is more uncertain and often oscillating.

TradingDa	12/24/07	12/26/07	12/27/07	12/28/07	12/31/07	01/02/08	01/03/08	01/04/08	01/07/08	01/08/08	01/09/08
DaystoExp	36	35	34	33	32	31	30	29	28	27	26
Interest	0.064786	0.011063	0.042798	0.044095	0.054629	0.034395	0.009678	0.033083	0	0.103021	0.041009
StockPrice	28.72	28.38	27.79	27.56	27.0699	26.54	26.75	26.12	26.13	25.43	26.24
10	0	0	0	0	10	0	0	10	0	10	0
12.5	0	0	0	0	4	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0
17.5	0	0	0	0	0	0	0	46	0	0	47
20	0	0	0	10	22	23	190	306	131	1001	421
22.5	3	0	20	0	22	37	287	284	32	91	13
25	0	358	186	314	925	1398	924	3575	1985	3627	1760
27.5	242	1092	1065	5894	1822	2069	5663	4697	2594	11118	2121
30	470	1185	2497	1479	1634	1651	2050	1504	952	4321	495
32.5	531	995	79	592	178	206	182	47	120	0	30
35	137	0	5	2	0	0	0	0	0	0	0
37.5	0	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0	0
42.5	0	0	0	0	0	0	0	0	0	0	0
45	0	0	0	0	0	0	0	0	0	0	0

From this table, it can be seen that most of the trading volume occurs near the moneyness that is close to 1. For the rest of the trading, there is very thin or even no volume.

TradingDa	12/19/06	12/20/06	12/21/06	12/22/06	12/26/06	12/27/06	12/28/06	12/29/06	01/03/07	01/04/07	01/05/07	01/08/07
DaystoExp	40	39	38	37	36	35	34	33	31	30	29	28
Interest	0.065637	0.064813	0.066518	0.05776	0.070212	0.069427	0.037077	0.049993	0.041292	0.045261	0.026724	0.051967
StockPrice	27.63	27.39	27.29	26.93	27.19	27.3	27.42	27.33	27.73	28.46	28.47	28.63
10	0.2	0.2	0.1	0.2	0.1	0.2	0.2	0.1	0.2	0.2	0.2	0.1
12.5	0.1	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.1	0.1
15	0.1	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.1	0.1	0.2
17.5	0.2	0.1	0.2	0.2	0.2	0.1	0.2	0.2	0.2	0.1	0.2	0.2
20	0.2	0.1	0.1	0.2	0.1	0.1	0.1	0.1	0.1	0.1	0.2	0.2
22.5	0.2	0.1	0.1	0.2	0.1	0.1	0.1	0.2	0.2	0.2	0.1	0.1
25	0.1	0.05	0.05	0.1	0.1	0.1	0.1	0.1	0.2	0.1	0.2	0.1
27.5	0.05	0.05	0.05	0.05	0.05	0.1	0.05	0.05	0.1	0	0.1	0.05
30	0.1	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
32.5	0.05	0.05	0.05	0.05	0.05	0.1	0.1	0.05	0.05	0.05	0.05	0.1
35	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
37.5	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05
40	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05

From this table, it can be seen that when the moneyness goes closer to 1, the bid-ask spread gets narrower, when the moneyness goes farther away from 1, the bid-ask spread gets wider.

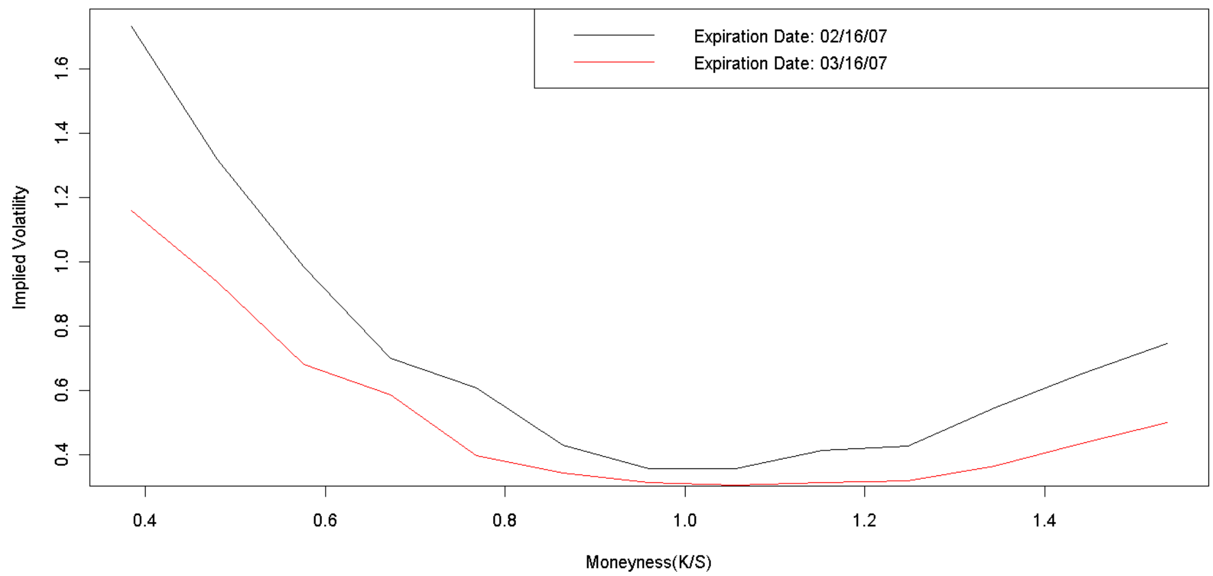
Here is another example of the trading volume and bid-ask spread, similar pattern can be observed.

TradingDa	01/22/07	01/23/07	01/24/07	01/26/07	01/29/07	01/30/07	01/31/07	02/01/07	02/02/07	02/05/07	02/06/07	02/07/07
DaystoExp	38	37	36	34	33	32	31	30	29	28	27	26
Interest	0.007004	0.050513	0.027146	0.039222	0.018831	0.011091	0.034395	0.020714	0.018364	0.009505	0.013145	0.023901
StockPrice	30.72	30.74	31.09	30.6	30.53	30.48	30.86	30.56	30.19	29.61	29.51	29.37
10	0	0	0	0	0	0	0	0	0	0	0	0
12.5	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0
17.5	0	0	0	0	0	0	0	0	29	0	0	0
20	0	0	0	0	0	0	0	0	0	20	0	0
22.5	0	0	0	250	0	0	0	0	120	10	0	0
25	0	1	294	57	16	0	0	0	11	6	40	50
27.5	3	430	188	23	26	28	511	55	186	533	187	201
30	18	1783	549	3302	1254	494	1540	2997	4598	4379	4716	6946
32.5	4130	825	5591	9603	3279	1437	3129	2658	4360	4995	400	2855
35	3	0	50	4090	94	366	160	82	1630	0	0	0
37.5	0	0	0	0	0	0	0	0	5	0	0	0
40	0	0	0	0	0	0	0	0	0	0	0	0
42.5	0	0	0	0	0	0	0	0	0	0	0	0

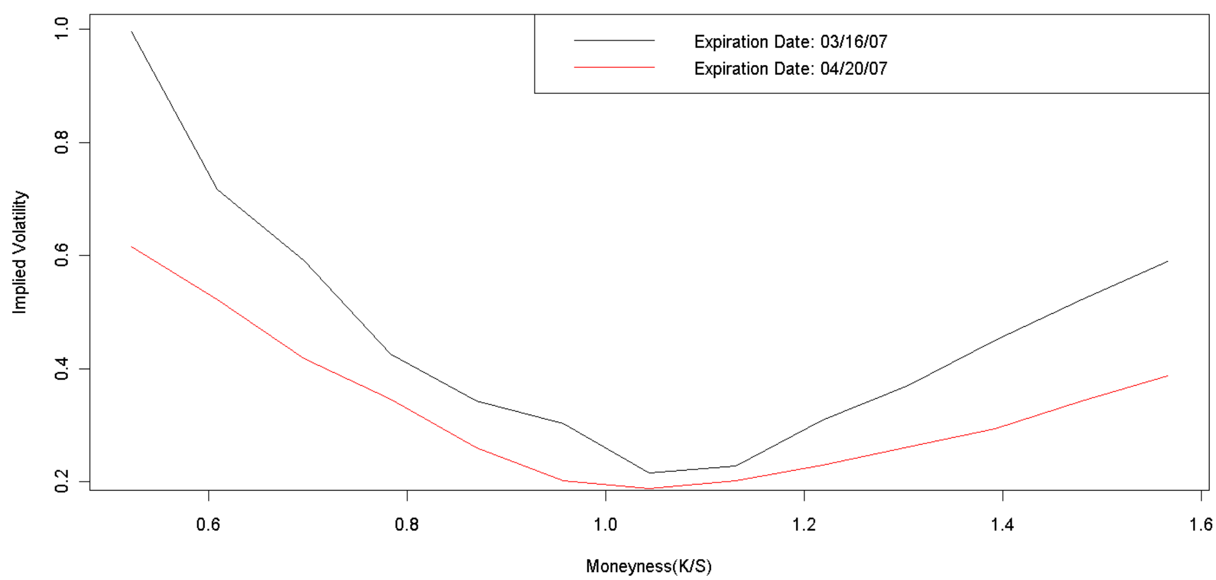
TradingDa	01/22/07	01/23/07	01/24/07	01/26/07	01/29/07	01/30/07	01/31/07	02/01/07	02/02/07	02/05/07	02/06/07	02/07/07
DaystoExp	38	37	36	34	33	32	31	30	29	28	27	26
Interest	0.007004	0.050513	0.027146	0.039222	0.018831	0.011091	0.034395	0.020714	0.018364	0.009505	0.013145	0.023901
StockPrice	30.72	30.74	31.09	30.6	30.53	30.48	30.86	30.56	30.19	29.61	29.51	29.37
10	0.2	0.2	0.2	0.1	0.1	0.2	0.2	0.1	0.05	0.1	0.1	0.1
12.5	0.1	0.2	0.2	0.1	0.1	0.2	0.2	0.1	0.1	0.1	0.1	0.1
15	0.1	0.2	0.2	0.1	0.1	0.2	0.2	0.1	0.1	0.1	0.05	0.1
17.5	0.1	0.2	0.2	0.2	0.1	0.1	0.2	0.1	0.1	0.1	0.05	0.1
20	0.2	0.2	0.2	0.2	0.1	0.1	0.2	0.1	0.05	0.1	0.05	0.1
22.5	0.2	0.1	0.1	0.2	0.2	0.1	0.1	0.1	0.05	0.1	0.1	0.1
25	0.2	0.1	0.1	0.2	0.2	0.1	0.1	0.2	0.1	0.1	0.1	0.05
27.5	0.1	0.1	0.1	0.2	0.2	0.1	0.2	0.1	0.04	0.04	0.04	0.04
30	0.1	0.1	0.05	0.1	0.1	0.05	0.05	0.05	0.01	0.03	0.02	0.02
32.5	0.05	0.05	0.05	0.1	0.05	0.05	0.05	0.05	0.01	0.02	0.02	0.02
35	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.02	0.02	0.02	0.02
37.5	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.02	0.02	0.02	0.02
40	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.01	0.01	0.01	0.01

For the stock option trading that carries the same date with different expiration dates, it can be seen that the implied volatility vs. moneyness curve is very different. As the option gets closer to the expiration date, the volatility becomes higher with also higher open interest. The pattern can be observed with the following sample Cisco and Microsoft option trading curves:

CISCO European Call Option Implied Volatility vs. Moneyness (Trade Date: 01/23/07)



MSFT European Call Option Implied Volatility vs. Moneyness (Trade Date: 02/16/07)



6. Appendix

The following expiration data lists all expiration dates covered by the dataset. To generate the expiration dates file used by the utility program, copy the data and save it into a text file.

01/21/05 02/18/05 04/15/05 03/18/05 07/15/05 05/20/05 06/17/05
08/19/05 09/16/05 10/21/05 11/18/05 12/16/05 01/20/06 02/17/06
03/17/06 04/21/06 05/19/06 06/16/06 07/21/06 08/18/06 10/20/06
0915/06 11/17/06 12/15/06 01/19/07 02/16/07 03/16/07 04/20/07
05/18/07 06/15/07 07/20/07 08/17/07 09/21/07 10/19/07 11/16/07
12/21/07 01/18/08 02/15/08 03/21/08 04/18/08 05/16/08 06/20/08
07/18/08 08/15/08 09/19/08 10/17/08 11/21/08 12/19/08 01/16/09
02/20/09 03/20/09 04/17/09 05/15/09 06/19/09 07/17/09 08/21/09
09/18/09 10/16/09 11/20/09 12/18/09

The following stock symbol data lists S&P 100 stock symbols covered by the dataset. To generate the S&P 100 stock symbol file used by the utility program, copy the data and save it into a separate text file.

AA AAPL ABT AEP ALL AMGN AMZN AVP AXP BA BAC BAX BHI BK BMY
BRK.B CAT C CL CMCSA COF COP COST CPB CSCO CVS CVX DD DELL DIS
DOW DVN EMC ETR EXC F FCX FDX GD GE GILD GOOG GS HAL HD HNZ
HON HPQ IBM INTC JNJ JPM KFT KO LMT LOW MA MCD MDT MET MMM
MO MON MRK MS MSFT NKE NOV NSC NWSA NYX ORCL OXY PEP PFE PG PM
QCOM RF RTN S SLB SLE SO T TGT TWX TXN UNH UPS USB UTX VZ WAG
WFC WMB WMT WY XOM XRX

7. References

1. WPI research report “Restructuring Option Chain Data Sets Using Matlab” written by Alison Wooden in May 2010.

2. Shreve, Steven E. 2004, “Stochastic Calculus for Finance II: Continuous-Time Models”, Springer, 2004.

3. John Hull, “Option, Futures and Other Derivatives”, 6^{th} Edition, Pearson Prentice Hall, 2006.